

仮想現実感による斜面滑走シミュレータの動作改良

Behavior improvement of virtual reality slope gliding simulator

○奥山瞬¹, 山口健², 吉川浩²* Shun Okuyama¹, Takeshi Yamaguchi², Hiroshi Yoshikawa²

Abstract: In our previous study, it was only possible to move the character by balancing on the Wii Balance Board but the orientation of the character was never changed. With the help of Unity it was possible to improve the game to make character rotation possible. When the player changes balance the player's character is turned automatically into the direction of its movement vector.

1. まえがき

本研究では統合型ゲーム開発環境である Unity^[1] を使用し, VR タライスキー(以下タライスキー) の製作及びゲーム性の向上を行っている^[2]. 現在のタライスキーは, Wii バランスボードを使用し, プレイヤは重心を変化させ, キャラクタの操作を行っている. しかし, 従来のタライスキーは進行方向に関係なくキャラクタが常に同一の方向に向いており, 重力の斜面接線成分(以下斜面方向)とキャラクタの向きが一致しないときは, 横滑りが発生していた.

本報告では, 本来の物理挙動を行えるようにキャラクタの動作を改善する.

2. Unity を用いたタライスキーの概要

Fig. 1 にタライスキーの概要を示す. タライスキーとは障害物を避けながら斜面を滑走し, ゴールを目指すスキーシミュレーションである. キャラクタはシミュレーションを行う際に使用する物体, プレイヤは操作をする人物とする. プレイヤは wii バランスボードを使用し, 重心を変化させ, キャラクタを操作する. タライスキーにはタイトル画面とスキーシミュレーションの画面と結果の画面をそれぞれ用意する. Fig. 2 にそれぞれの画面を示す. タイトル画面では Fig. 2 のようにモードの選択を行う. スキーシミュレーションの画面ではキャラクタを操作し, 障害物を避けながら斜面を滑走し, ゴールを目指す. 結果の画面ではスタートからゴールまでにかかった時間を表示する. 表示されたタイムを保存し, 保存されたタイムの中から時間が短いもの順に表示させるランキングがある.

3. キャラクタの回転と当たり判定

キャラクタの向きが変化しない問題を解決するため, キャラクタの底につけた 2 枚の板状の当たり判定の摩擦係数を変更出来るようにし, トルクによる回転を行うようにする. キャラクタのオブジェクトと違う当たり判定になる理由として, 従来の 1 枚の板状の当たり判定だとトルクによる回転を行えないためであ

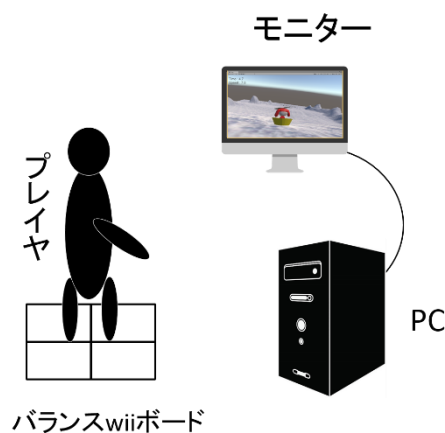


Figure 1. Tub Skiing of Overview Figure



Figure 2. Each of the screen

る. このとき, トルクによってできるキャラクタの角度を算出する式を式(1)に示す.

$$\theta = \frac{g}{r} (\sin\omega - \mu\cos\omega) \Delta t^2 \quad (1)$$

ここで, キャラクタの角度を θ , キャラクタが触れている斜面の角度を ω , 回転にかかる時間を Δt , 重力加速度を g , 一定の動摩擦係数を μ , キャラクタから板状の当たり判定までの距離を r とおく. 式(1) による回転の例を Fig. 3 に示す.

4. キャラクタの力の制御

実際のスキーの動作では, エッジを効かせて横滑りを抑制し, 体の向いている方向に進んでいくことができる. タライスキーではキャラクタが曲がった際にそ

の正面方向に進むように横滑りを抑える力を加える。
 キャラクタに摩擦をかけ回転した際、Fig. 4 のように斜面に対して横を向いた際にも斜面方向に滑り続ける。キャラクタに重力と摩擦以外の力が加わっていないため、キャラクタが斜面方向に対して回転してもその方向には進まない。そのため、キャラクタが回転した際に、キャラクタの正面方向に進むようにする。Fig. 4 にキャラクタにかかる力の加わり方、左右の力の制御のフローチャートを Fig. 5 に示す。Fig. 5 にあるように判定を取り、制御を行う。キャラクタにかかる力を F 、重力加速度を g 、キャラクタの質量を m 、キャラクタが触れている斜面の角度を ω 、斜面方向のベクトルとキャラクタ正面方向のベクトルの角度を θ とおき、式(2)で表すことができる。

$$|F| = mg \cos \omega \sin \theta (0 < \theta < \pi) \quad (2)$$

ただし、キャラクタの進行方向の角度と斜面の方向の角度を算出した際、キャラクタがどの方向を向いていても鋭角の値を取得する。そのため、キャラクタが斜面方向に対して、左右のどちらの方向を向いているか判定することができない。Fig. 4 の斜面方向のベクトルの終点座標とキャラクタの正面方向のベクトルの終点座標を取得し、取得した座標の関係を用いて、キャラクタの左右の判定をとる。このとき、キャラクタに力が加わるため速度が変化する。変化した際の速度の式を式(3)に示す。

$$V = g \cos \omega \sin \theta \Delta t + \Delta V \quad (0 < \theta < \pi) \quad (3)$$

V はキャラクタの速度ベクトル、 Δt は単位時間とする。 ΔV は Δt 秒前の速度とする。

5. 結果

キャラクタの 2 枚の当たり判定の摩擦の値を変更しトルクを使用することでキャラクタを回転させるようにした。

斜面方向に対して、キャラクタがどちらを向いているかの判定を行い、キャラクタに常に横滑りを抑制する力が加わるようにした。Fig. 6 に実装後の軌跡を示す。Fig. 6 から、キャラクタが回転を行った際に曲がる動作が行われることがわかる。

6. むすび

本研究では、Unity を用いて従来のタライスキーよりも没入感があり、より実際の物理挙動に近いタライスキーの作製するための制御方法についていくつか検証を行った。

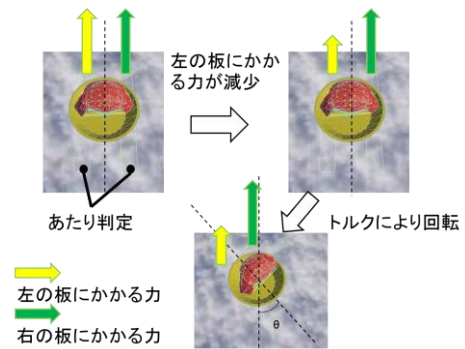


Figure 3. Character's Collision detection

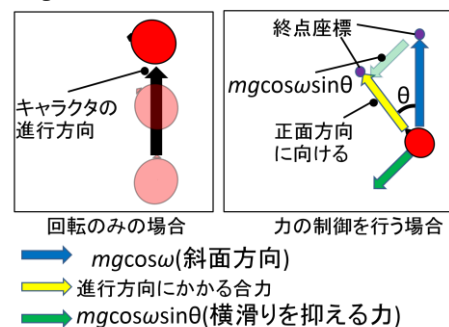


Figure 4. The resultant force of the force

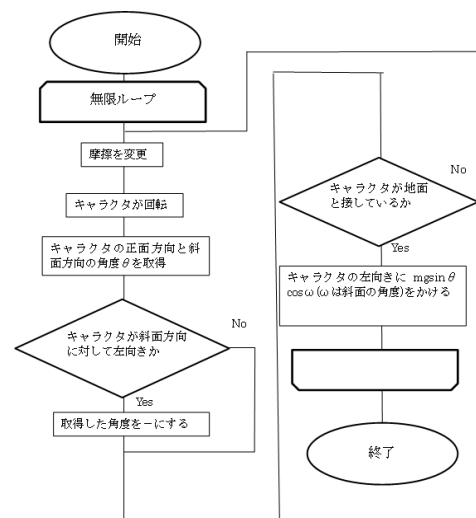


Figure 5. Flow chart of the control of the force

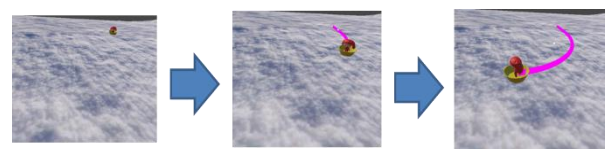


Figure 6. The trajectory of the character

7. 参考文献

- [1] Unity, "<http://japan.Unity3d.com/>"
- [2] 黒川 勝貴: "Unity を用いたタライスキーのゲーム性向上", 2015 年度日本大学理工学部電子情報工学科卒業論文 (2015)
- [3] vive ホーム, "<https://www.htcvive.com/jp/>"